Using Git & Github for Collaborative Development

Recitation 2, CMU 17-313, Fall 2021

Overview:
After this recitation, the students should be able to use Git and GitHub for their teamwork in this course and other projects. Students should know about the GitFlow workflow and how to collaborate with other developers in their projects.

Prerequisites:

- You have a GitHub account
- You have Git installed
- You know Git Basics

You would be doing the activities as pairs! Find a partner. Unless specified, each step is to be done by both person

Activity 1: Forking a repo and adding a collaborator

- You or your partner forks this repository, then add the other person as a collaborator on GitHub
- Clone the repository on your local machine
- Run the python code and figure out the two bugs in the code (the print statements should point you to the bugs that is there)

Activity 2: Setting up Kanban board and creating issues

- On GitHub, under the project tab, one person creates a new project. Select "Automated Kanban" as the template
- Create two issues, one for each of the bug found.
- Add the two issues to the Kanban to-do board
- Assign both issues, one for each person
- Move the issues into To-do!

Activity 3: Branching, committing, and pushing

- In your local machine, create a new branch with a relevant name to the issue you are addressing (e.g. *"fix-header-sizing-issue", "fix-multiple-dialog-bug", "add-calendar-feature")*
- Fix the issue that you were assigned, then add and commit the changes
    - Commits should start with a verb, and what it does to the codebase (e.g. *"Remove faulty condition from getCustomerDetails", "Fix failing CompositeTestCase")*
- Push the branch to remote

Activity 4: Writing pull requests

- On GitHub, create a pull request to merge the changes from the branch you have just pushed to the master branch.
    - Name your pull request appropriately
    - In the description, describe what changes has been made to address the issue, and how has the changes been tested
- Assign the issue you created to the pull request
- Request your partner for a code review!

Activity 5: Doing code reviews and resolving merge conflicts

- Review your partner's code (use the review change button). Approve changes if it looks good
- Once your PR has been reviewed, merge your changes! Resolve any merge conflicts that arises accordingly
- Return to the kanban board, and if you have done things right the issue should automatically move into Done